



Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

Bescheinigung

Certificate

Attestation

Die angehefteten Unterla-
gen stimmen mit der
ursprünglich eingereichten
Fassung der auf dem näch-
sten Blatt bezeichneten
europäischen Patentanmel-
dung überein.

The attached documents
are exact copies of the
European patent application
described on the following
page, as originally filed.

Les documents fixés à
cette attestation sont
conformes à la version
initialement déposée de
la demande de brevet
européen spécifiée à la
page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

99480001.9

**CERTIFIED COPY OF
PRIORITY DOCUMENT**

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

Adette Fiedler

A. Fiedler

DEN H
THE P
LA H

08/03/99

JC662 U.S. PTO

09/493242



01/28/00

BLANK PAGE



Europäisches
Patentamt

European
Patent Office

Office européen
des brevets

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:
Demande n°: 99480001.9

Anmeldetag:
Date of filing:
Date de dépôt: 29/01/99

Anmelder:
Applicant(s):
Demandeur(s):
INTERNATIONAL BUSINESS MACHINES CORPORATION
Armonk, NY 10504
UNITED STATES OF AMERICA

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
File manager for storing several versions of a file

In Anspruch genommene Priorität(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat:
State:
Pays:

Tag:
Date:
Date:

Aktenzeichen:
File no.
Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:

/

Am Anmeldetag benannte Vertragsstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE
Etats contractants désignés lors du dépôt:

Bemerkungen:
Remarks:
Remarques:

BLANK PAGE

FILE MANAGER FOR STORING SEVERAL VERSIONS OF A FILE***Technical Field***

5

The present invention relates to computer systems and more particularly to a file manager for managing a plurality of files when it is required to store several versions for said files.

Background Art

15 In a computer system, a file manager shows where a file which is commonly identified by a path is physically located. Such a path is composed of a logical unit, a directory and a file name. The logical unit is generally referenced by a capital letter followed by a colon, C: for instance. A directory gathers several files that have common characteristics. A classification is thus defined for the plurality of files linked to a same logical unit.

20 Furthermore, a directory can be subdivided in subdirectories themselves subdivided in sub-subdirectories and so on. The word directory will now be used with its generic meaning, that is to say that it refers to the whole hierarchy of the classification.

25 Finally, a file has a file name, FILE1 for instance. When the file FILE1 belongs to the directory DIR1 linked to logical unit C:, it is identified by the path C:\DIR1\FILE1. The directory and the file name represent an identifier which means that the file FILE1 is identified by a logical unit C: and the identifier \DIR1\FILE1.

The file manager therefore indicates that the file C:\DIR1\FILE1 is on a definite physical unit 3X, a hard disk drive for instance.

30 Now let us assume that the computer is provided with a standard software such as a word processing software. Naturally, all the files of this software are known by the file manager.

Then, let us furthermore assume that a new version of this standard software should be implemented in the computer. This new version comprises files that have the same name as files of the previous version. According to a first solution, the operating system of the computer will store such a file of the new version in the file
5 of the previous version, since they have the same identification. When these two files are different, there could be a malfunction when operating a program or an application file that needs the previous version of the standard software.

According to a second solution, the operating system of the computer will change the identification of the new file in order not to delete the previous one. As
10 an example, the file SETUP of the previous version was identified as F:\INSTALL\SETUP and the corresponding file of the new version is now identified as F:\SETUP\SETUP. If a third version of the same standard software happens to be installed, this could lead to another file identified for instance F:\INSTALL\SETUP\SETUP. When these three files are identical, there is a useless
15 duplication that leads to an increase in the required memory for storing the software. It is actually not possible to delete one of these files since some programs or application files know only F:\INSTALL\SETUP, other ones know only F:\SETUP\SETUP and the most recent ones know only F:\INSTALL\SETUP\SETUP.

20

Summary of the Invention

It is therefore a general object of the present invention to provide for a file manager that keeps in records all the files required by a computer and that allows to avoid any duplication.

25 In accordance with the present invention, a file manager is provided for locating a file identified by a path referring to a logical unit and an identifier, the file manager comprising a table associating the file with a priority list of physical units.

So, several versions of a file with a common logical identification can be saved.

30 In order to have access to any one of these versions, the file manager comprises means for modifying the priority list.

Preferably, the table also defines attributes (Read, Write, Hide, Execute) for each item of the priority list.

In this case, according to an advantageous feature, the table defines for the file a physical unit with a write attribute that differs from a physical unit with a read
5 attribute.

In addition, the file manager comprises means for modifying the attributes.

In a preferred embodiment, the file manager is incorporated into the controller of peripheral devices in a computer system.

Alternatively, the file manager is incorporated into the operating system of a
10 computer system.

Brief description of the Drawings

Embodiments of the present invention are described below by way of
15 example only with reference to the accompanying drawings, wherein :

Figure 1 represents the table of a file manager according to the present invention;

Figure 2 represents the location of a file manager according to a preferred
20 embodiment of the invention.

Detailed Description of the Invention

The invention applies to a single personal computer or to a workstation, but it
25 also applies to a server of a network connecting several computers.

Besides, files stored in a computer or a server are often provided with attributes generally referred to as follows:

- R for read, which means that this file can be read,
- W for write, which means that this file can be written or modified,
- 30 - H for hide, when the involved file should not be accessed,
- X for execute, when the involved file is a program that can be executed.

Now let us consider the file C:\DIR1\FILE1 of a standard software. A first version V1 of this file is stored on physical unit 1Z located in a server.

Later on, an updated version of the standard software is commercialized and a second version V2 of file C:\DIR1\FILE1 is consequently stored by a first network user on physical unit 2Y which is for instance the hard disk drive of his personal computer connected to the server.

Still later on, the updated software is put at the disposal of every network user and consequently, a third version V3 of file C:\DIR1\FILE1 is stored on physical unit 3X in the server.

10 With reference to Figure 1, the file manager of the first user comprises a table which shows that this file has an X attribute and is associated with a priority list of physical units 3X, 2Y, 1Z respectively corresponding to versions 3, 2 and 1 of this file.

When the first user starts an execution of file FILE1, the file manager will at first select the corresponding file in physical unit 3X. If this file is not available, the file manager will then look for the version stored in physical unit 2Y and if this version is not available either, it will chose the first version stored in physical unit 1Z.

Between the installation of versions 2 and 3 of file FILE1, the priority list is naturally limited to 2Y, 1Z. An execution of file FILE1 is therefore operated with version 2. If a malfunction appears with a program or an application file created prior to the installation of version 2, it is necessary to come back to version 1 of file FILE1. This operation is easily done by modifying the priority list such as it becomes 1Z, 2Y.

After the installation of version 3 of file FILE1, the priority list is thus 3X, 2Y, 1Z. It will be appreciated that if a problem occurs with the third version of file FILE1, it is possible to come back to version 2 of this file with a modification of the priority list. Furthermore, when it is sure that versions 2 and 3 are equivalent, it is possible to remove version 2 from the priority list which becomes 3X, 1Z. It is even possible to delete version 2 in physical unit 2Y, which spares memory in this unit.

The equivalence between versions 2 and 3 can be established by different means. A first way is to check that the sizes and parities of both versions are equal.

A second way, more empirical, consists in stating that if no problem has occurred with version 3 during a definite lapse of time, version 3 is then satisfactory.

The invention will also find an advantageous application in the management of a software that is shared by a plurality of users. Classically, such a software, located in a server, is composed of common files and customization files. A common file is common for all users, which means that such a file is stored only once. A customization file is specific to an identified user which means that such a file is stored a plurality of times corresponding to the number of users under a corresponding plurality of identifications.

Naturally, some of the common files need customization files to run. It then appears that the software should comprise selection means to designate which customization file should be used by such a common file.

The invention allows to delete these selection means since a customization file can be designated by a generic identification, C:\DIR1\FILE4 for instance. The selection is then made by the file manager.

As an example, the file manager associates this customization file respectively with physical unit 4X for user A, 4Y for user B, 4Z for user C and so on. It should further be noted that the customization file could be associated with the same physical unit provided that this unit designates a drive belonging to a personal computer and not to the server.

The invention will find another advantageous application for giving different attributes to files identified by the same path but associated with different physical units. As an example, the file identified C:\DIR1\FILE2 with a read attribute has a priority list 3Z, 2Y and this same file with a write attribute has a priority list 2Y, 3Z.

Accordingly, this file is read in physical unit 3Z and when it is modified, it is stored in physical unit 2Y.

It follows that two versions of the same file are available, the old one on 3Z and the new one with the modifications on 2Y.

Usually, the new version, which is saved on 2Y, is the version required for further use. It is then necessary to reverse the priority list for the read attribute which

then becomes 2Y, 3Z in order to be able to read the new version. Consequently, the priority list for the write attribute should also be reversed.

Nevertheless, when reading the new version, it could appear that this version is erroneous because the modifications of the old version should not have been made. It is therefore suitable to restore the old version and this is done by reversing again both priority lists which recover their initial states, 3Z, 2Y for the read attribute and 2Y, 3Z for the write attribute.

Considering only the first item of each priority list, it appears that the physical unit with a write attribute differs from the physical unit with a read attribute.

10

With reference to Figure 2, there is shown where the file manager according to a preferred embodiment is located. The file manager is implemented in an equipment such as a personal computer, a workstation or a server.

The equipment essentially comprises a central processor unit CPU and a memory RAM, each one connected to a first controller CONT. Controller CONT provided for controlling the bus and the memory is further connected to an adapter ADA by a bus generally referred to as PCI (Peripheral Component Interconnect) bus.

The adapter ADA is preferably a SCSI adapter, where SCSI stands for Small Computer System Interface which is an ANSI standard. This standard defines a bus and the logical interfaces associated to this bus for interconnecting computers and peripheral devices. This adapter is therefore provided with a PCI interface connected to the PCI bus and to a second controller MASTER in charge of controlling the peripheral devices. This second controller is also connected by a SCSI bus to an internal interface INT and to an external interface EXT. The disk drives of the computer are connected to the internal interface while the external interface is used for external connection, to a network for instance.

According to this preferred embodiment, the file manager FM is incorporated to the second controller MASTER.

Alternatively, the file manager can be implemented as a function of the operating system of the computer. Thus, the table, which contains the location and

attributes of each file, is stored in the hard disk drive. This table may be duplicated in the memory RAM for performance improvement.

The scope of the present invention is in no way limited to the above
5 embodiments. In particular, any means could be replaced by equivalent means.

BLANK PAGE

CLAIMS

1. File manager (FM) provided for locating a file identified by a path referring to a logical unit (C:\) and an identifier (\DIR\FILE1), characterized in that it comprises a table associating said file (C:\DIR1\FILE1) with a priority list of physical units (3X, 2Y, 1Z).
2. File manager (FM) according to claim 1, further comprising means for modifying said priority list (3X, 2Y, 1Z).
3. File manager (FM) according to claim 1 or 2, wherein said table also defines attributes (R, W, H, X) for each item of said priority list (3X, 2Y, 1Z).
4. File manager (FM) according to claim 3, wherein said table defines for said file a physical unit with a write attribute (2Y) that differs from a physical unit with a read attribute (3Z).
5. File manager (FM) according to claim 3 or 4, further comprising means for modifying said attributes.
6. File manager (FM) according to any one of the preceding claims, incorporated into the controller (MASTER) of peripheral devices in a computer system.
7. File manager according to any one of claims 1 to 5, incorporated into the operating system of a computer system.
8. Medium comprising computer readable instructions for carrying out a file manager (FM) according to any one of the preceding claims.

BLANK PAGE

FILE MANAGER FOR STORING SEVERAL VERSIONS OF A FILE***Abstract of the Disclosure***

5

A file manager is provided for locating a file identified by a path referring to a logical unit (C:\) and an identifier (\DIR\FILE1). In accordance with the invention, the file manger (FM) comprises a table associating the file (C:\DIR1\FILE1) with a priority list of physical units (3X, 2Y, 1Z). In a preferred embodiment, the file manager (FM) is incorporated into the controller (MASTER) of peripheral devices in a computer system.

(Fig. 1)

15

BLANK PAGE

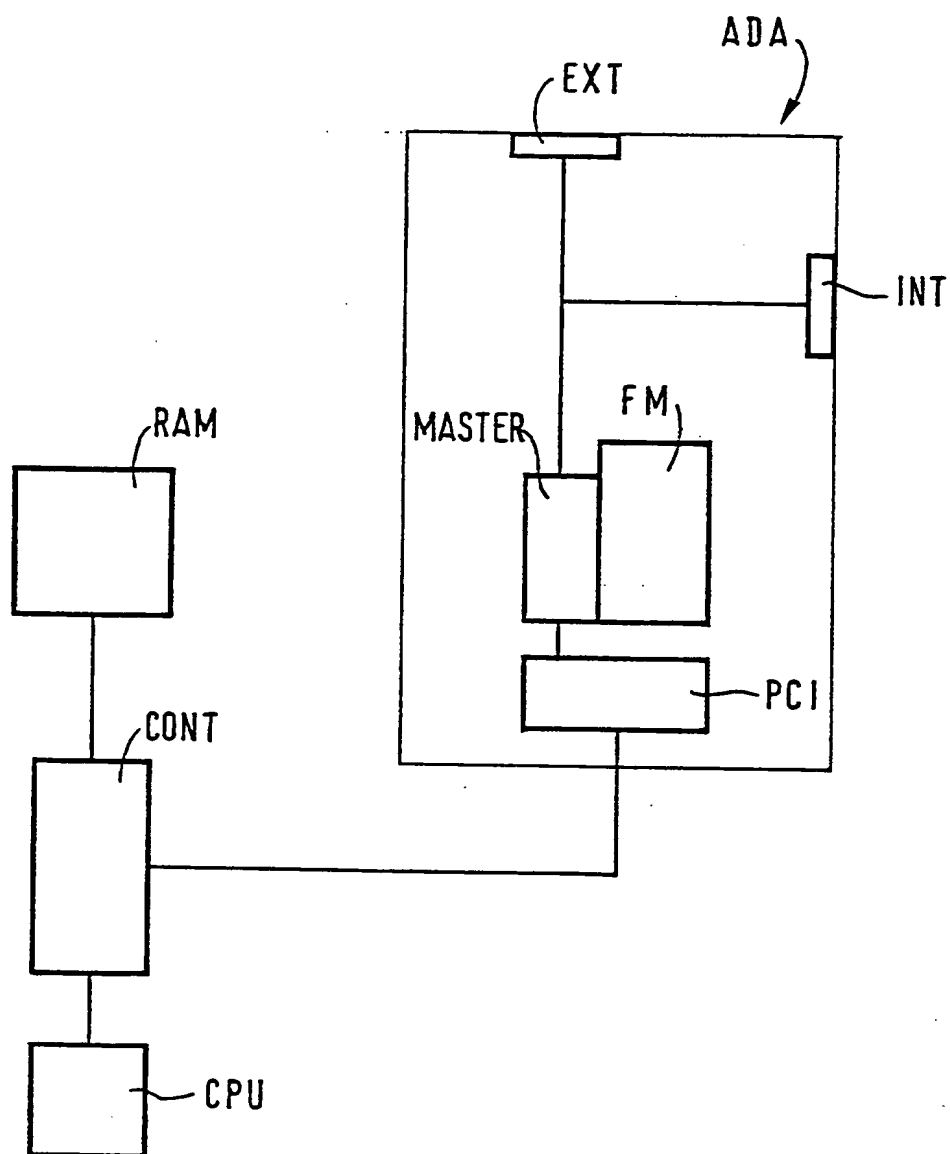
1/2

IDENTIFICATION	ATTRIBUTE	PRIORITY LIST
C:		3Z
\DIR1		3Z
\FILE1	X	3X, 2Y, 1Z
\FILE2	R	3Z, 2Y
	W	2Y, 3Z
\FILE3	H	
\FILE4	R	4X
\FILE5	H	
\DIR2		3Z
D:		1X

Figure 1

2/2

FIG. 2



Priority Document

1 of 1

Inventor: Benayoun et al. Dkt No.: FR998-073
Serial No.: Not yet assigned Filed: January 28, 2000

FILE MANAGER FOR STORING...FILE

McGinn & Gibb, P.C.

Customer No. 21254

BLANK PAGE